UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 10/829,591 | 04/21/2004 | Matthew Conover | SYMC1050 | 6738 |

34350 7590 06/03/2008
GUNNISON, MCKAY & HODGSON, L.L.P.
1900 GARDEN ROAD, SUITE 220
MONTEREY, CA 93940

| EXAMINER | |
|---|---|
| PICH, PONNOREAY | |

| ART UNIT | PAPER NUMBER |
|---|---|
| 2135 | |

| MAIL DATE | DELIVERY MODE |
|---|---|
| 06/03/2008 | PAPER |

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

| Office Action Summary | Application No. | Applicant(s) | |
|---|---|---|---|
| | 10/829,591 | CONOVER, MATTHEW | |
| | Examiner | Art Unit | |
| | PONNOREAY PICH | 2135 | |

*-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --*

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1)☒ Responsive to communication(s) filed on *11 April 2008*.

2a)□ This action is **FINAL**. 2b)☒ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4)☒ Claim(s) *1-25 and 35-37* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-25 and 35-37* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

**Application Papers**

9)□ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *24 April 2008* is/are: a)☒ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1) ☒ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application

6) □ Other: _____.

## DETAILED ACTION

Applicant's arguments and amendments made in response to the election/restriction requirement were fully considered. The election/restriction requirement is withdrawn due to applicant's amendments. Claims 1-25 and 36-37 were examined and are pending.

### *Claim Objections*

Claims 4 and 11-12 are objected to because of the following informalities:

1. Claims 4 and 11 refers to an ObReferenceObjectByHandle() function. The specification of the current application does not define the algorithm that is meant to be encompassed by this particular function, thus the metes and bounds of the claim cannot be determined.
2. Claim 12 refers to an ObDereferenceObject() function. The specification of the current application does not define the algorithm that is meant to be encompassed by this particular function, thus the metes and bounds of the claim cannot be determined.
3. Appropriate correction is required.

### *Claim Rejections - 35 USC § 101*

35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claim 35 is rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

The system of claim 35 is not statutory because the claimed system is directed towards software per se, which does not fall within any of the four statutory categories of invention and software by itself is unable to produce any result. Claim 35 invokes 112, 6th paragraph and according to the specification, it would appear that each of the recited means refers to “a behavior blocking and monitoring application”, which is software per se. See also claim 36 as originally filed, which provides further evidence that the claimed means of claim 35 are software per se--"a behavior blocking and monitoring application".

### *Claim Rejections - 35 USC § 103*

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1, 2, 5, 14-24, and 35-37 are rejected under 35 U.S.C. 103(a) as being unpatentable over Van Dyke et al (US 6,412,070) in view of Ho (US 5,615,373) in further view of Krishnaswami et al (US 6,618,735), herein referred to as Krish.

**Claims 1, 35, 36, and 37:**

As per claim 1, Van Dyke discloses determining whether an attempter that originated said access attempt is authorized to access said object, wherein upon a determination that said attempter is authorized to access said object, said method further comprises providing access to said object (col 5, lines 17-49; col 6, lines 25-44; and col 7, lines 13-44).

Van Dyke does not explicitly disclose stalling an attempt to reference an object and providing access to said object comprises saving at least a part of said object. However, Ho teaches stalling an attempt to reference an object (col 8, lines 12-26). Note that in Ho's invention, if a first party already has a lock to an object, an attempt to reference an object by a second party is stalled, i.e. placed in a queue. Further, Krish teaches providing access to an object and saving at least a part of said object (col 5, lines 29-37).

At the time applicant's invention was made, it would have been obvious to one of ordinary skill in the art to modify Van Dyke's invention such that it stalled an attempt to reference an object as per Ho's teachings because it would prevent multiple parties from editing the same file at the same time, which might lead to data inconsistency and lost work.

At the time applicant's invention was made, it would have been obvious to incorporate Krish's teachings within Van Dyke and Ho's combination invention according to the limitations recited in claim 1 such that upon determination that said attempter is authorized to access said object, said method further comprises saving at least part of said object. One skilled would have been motivated to save at least a portion of said object as per Krish's teachings because it would allow Van Dyke's invention to undo file changes if the application/user that made the attempt made an invalid file change (Krish: col 5, lines 51-59). Note that a person of ordinary skill would understand that just because an application/user is authorized to access an object does not necessarily mean that any change that is made to the object is valid. Krish's teachings would allow for a way to undo invalid changes, which would provide for better system security.

Claim 35 is directed towards a system comprising (software) means for implementing the method of claim 1 and is rejected for much the same reason discussed above for claim 1. Van Dyke and Krish are implemented via a computer system, thus the method of their invention being accomplished via computer software is obvious.

Claim 36 is directed towards a computer-program product comprising a tangible computer-readable storage medium containing computer program code comprising a behavior blocking and monitoring application for implementing the method of claim 1 and is rejected for much the same reasons discussed above for claim 1. Van Dyke and Krish are implemented via a computer system, thus the method of their invention being accomplished via computer program code contained in a tangible computer-readable storage medium is obvious.

Claim 37 is directed towards a computer system comprising a memory having stored therein a behavior blocking and monitoring application; and a processor coupled to said memory, wherein execution of said behavior blocking and monitoring application generates a method comprising the steps of claim 1. Claim 37 is rejected for much the same reasons discussed above for claim 1. Note that because Van Dyke and Krish's inventions are implemented via computer systems, a memory and processor as recited are inherent to their computer systems.

**Claim 2:**

Ho further discloses wherein upon a determination that said attempter is authorized to access said object, said method further comprises releasing said attempt (col 8, lines 12-36).

**Claim 5:**

Van Dyke further discloses wherein upon a determination that said attempter is not authorized to access said object, said method further comprising denying said attempt (col 6, lines 25-44 and col 7, lines 13-44).

**Claim 14:**

Van Dyke further discloses determining whether said attempt has occurred (col 5, lines 17-49; col 6, lines 25-44; and col 7, lines 13-44).

**Claim 15:**

Krish further discloses stalling an attempt to release said object (col 5, lines 60-61).

**Claim 16:**

Krish further discloses determining whether said object has changed (col 5, lines 52-59).

**Claim 17:**

As per claim 17, the limitation further recited therein is obvious to the teachings of Krish. It is noted that in Krish's invention, an attempt to release an object is stalled if it is determined that changes to a protected file was made and a write operation cannot be performed until later (col 5, line 60-col 6, line 1). A person of ordinary skill having ordinary creativity and common sense would have recognized that if it was determined that an object was not changed, then it would have been obvious to release said attempt to release said object since the only reason to stall is that changes could not be made to the object until later.

**Claim 18:**

Krish further discloses wherein upon determination that said object has changed, said method further comprising determining if said attempter is authorized to change said object (col 5, lines 51-56).

**Claim 19:**

Krish further discloses wherein upon determining that said attempter is authorized to change said object, said method further comprises releasing said attempt to release said object (col 5, line 60-col 6, line 1).

**Claim 20:**

Krish further discloses wherein upon a determination that said attempter is not authorized to change said object, said method further comprising restoring said object (col 5, lines 51-59).

**Claim 21:**

Krish further discloses wherein said restoring comprises replacing part of said object with a saved at least part of said object (col 5, lines 51-59).

**Claim 22:**

Krish further discloses:

1. Stalling an attempt to release said object originating from said attempter (col 5, lines 60-61).
2. Determining that said object has been changed by said attempter (col 5, lines 51-53).
3. Determining that said attempter did not have authority to change said object (col 5, lines 51-56).
4. Restoring said object; and releasing said attempt (col 5, line 51-col 6, line 15).

**Claim 23:**

Van Dyke further discloses wherein said attempter is a user on a computer system (col 5, lines 17-36).

**Claim 24:**

Van Dyke further discloses wherein said attempter is a process on a computer system (col 5, lines 17-36). Note that an application executing on a computer system is a process.

Claims 3-4 are rejected under 35 U.S.C. 103(a) as being unpatentable over Van Dyke et al (US 6,412,070) in view of Ho (US 5,615,373) in further view of Krishnaswami et al (US 6,618,735), herein referred to as Krish, in further view of Vossen et al (US 6,026,402).

**Claim 3:**

As per claim 3, Ho teaches wherein upon said releasing said attempt, granting access to an object (col 8, lines 12-37). However, Van Dyke and Ho do not explicitly teach upon said releasing, said method further comprising determining if access is granted using an access control list. However, Vossen teaches accessing an object by calling the function ObReferenceObjectByFileHandle (col 6, lines 38-40), which one of ordinary skill in the art should understand is a standard MSDN function which returns an object pointer and before returning the pointer checks an access control list to determine if the requestor should be granted access to the requested object.

At the time applicant's invention was made, it would have been obvious to one of ordinary skill in the art to further modify Van Dyke's invention so that access to an object is done using the ObReferenceObjectByFileHandle function. Doing so would make it such that upon releasing said attempt (i.e. as per Ho's teachings), access to an object is granted using an access control list. One skilled would have been motivated to use ObReferenceObjectByFileHandle to reference an object because it is a standard MSDN function for file access and many software developers use Microsoft's development library for software development.

**Claim 4:**

As per claim 4, the limitation of wherein upon said releasing said attempt an ObReferenceObjectByFileHandle() function is invoked is made obvious over Vossen's additional teachings (col 6, lines 38-40) because as discussed in the rejection of claim 3, it would have been obvious to use the ObReferenceObjectByFileHandle to access an object since it is a standard MSDN function for file access.

Claims 6-8 are rejected under 35 U.S.C. 103(a) as being unpatentable over Van Dyke et al (US 6,412,070) in view of Ho (US 5,615,373) in further view of Krishnaswami et al (US 6,618,735), herein referred to as Krish, in further view of Hollander et al (US 6,412,071).

**Claim 6:**

As per claim 6, Van Dyke does not explicitly disclose hooking object functionality. However, Hollander discloses hooking object functionality (col 3, lines 62-67 and col 6, lines 5-10). At the time applicant's invention was made, it would have been obvious to one of ordinary skill in the art to further modify Van Dyke's invention such that it further comprised the step of hooking object functionality. One skilled would have been motivated to do so because hooking object functionality would allow Van Dyke's invention to interrupt system calls (Hollander: col 2, lines 23-34) to determine whether the call is coming from a valid address space (Hollander: col 3, line 67-col 4, line 6), which would prevent certain types of attacks on the computer system, i.e. buffer overflow attacks.

**Claim 7:**

As per claim 7, said object functionality comprising functionality associated with creating, modifying, or closing said object is disclosed by Van Dyke (col 6, line 57-col 7, line 10).

**Claim 8:**

As per claim 8, Hollander further discloses wherein said hooking object functionality comprises hooking a user mode library (col 2, lines 24-44). User applications use user mode libraries.

Claims 9-13 are rejected under 35 U.S.C. 103(a) as being unpatentable over Van Dyke et al (US 6,412,070) in view of Ho (US 5,615,373) in further view of Krishnaswami et al (US 6,618,735), herein referred to as Krish, in further view of Hollander et al (US 6,412,071) in further view of Dabak et al ("Hooking Windows NT System Services").

**Claim 9:**

Hollander discloses hooking object functionality (col 3, lines 62-67 and col 6, lines 5-10), however does not disclose hooking a system call table. However, Dabak discloses hooking a system call table (p2, paragraph 3, i.e. System Service Dispatch Table). It would have been obvious to one skilled in the art to further modify the combination invention of Van Dyke, Krish, and Hollander such that hooking object functionality comprises hooking a system call table. One skilled would have been motivated to do so because the easiest way to put a hook into system services is by using a system call table/system service dispatch table as disclosed by Dabak (p2, paragraph 3).

**Claim 10:**

Hollander discloses hooking object functionality (col 3, lines 62-67 and col 6, lines 5-10), however does not disclose hooking an object manager. However, Dabak discloses hooking an object manager (p2, paragraph 3). The system service dispatch table can be considered an object manager. It would have been obvious to one skilled in the art to further modify the combination invention of Van Dyke, Krish, and Hollander such that hooking object functionality comprises hooking an object manager. One skilled would have been motivated to do so because the easiest way to put a hook into system services is by using an object manager/system service dispatch table as disclosed by Dabak (p2, paragraph 3).

**Claim 11:**

Hollander discloses hooking object functionality (col 3, lines 62-67 and col 6, lines 5-10), however does not disclose hooking an ObReferenceObjectByHandle() function. However, Dabak discloses the limitation (see source code spanning pages 3-7). At the time applicant's invention was made, it would have been obvious to hook an ObReferenceObjectByHandle() function because it would allow one to determine if the application which called the function is from a valid address space prior to executing the function to provide access to an object. This would prevent certain types of attacks, as discussed by Hollander (col 3, line 67-col 4, line 6).

**Claim 12:**

Hollander discloses hooking object functionality (col 3, lines 62-67 and col 6, lines 5-10), however does not disclose hooking an ObDereferenceObject () function. However, Dabak discloses the limitation (see source code spanning pages 3-7). At the time applicant's invention was made, it would have been obvious to hook an ObDereferenceObject () function because it would allow one to determine if the application which called the function is from a valid address space prior to executing the function to provide access to an object. This would prevent certain types of attacks, as discussed by Hollander (col 3, line 67-col 4, line 6).

**Claim 13:**

Hollander discloses hooking object functionality (col 3, lines 62-67 and col 6, lines 5-10), however does not disclose hooking object type procedures. However, Dabak discloses the limitation (see source code spanning pages 3-7). At the time applicant's invention was made, it would have been obvious to hook object type procedures because it would allow one to determine if the application which called the function is from a valid address space prior to executing the function to provide access to an object. This would prevent certain types of attacks, as discussed by Hollander (col 3, line 67-col 4, line 6).

Claim 25 is rejected under 35 U.S.C. 103(a) as being unpatentable over Van Dyke et al (US 6,412,070) in view of Ho (US 5,615,373) in further view of Krishnaswami et al (US 6,618,735), herein referred to as Krish, in further view of Treadwell, III et al (US 5,845,280).

**Claim 25:**

Van Dyke further discloses wherein said attempter is a process on a computer system (col 5, lines 17-36). Note that an application executing on a computer system is a process. Van Dyke is silent with respect to the process being a kernel mode process.

However, processes being kernel mode process was disclosed by Treadwell (col 2, lines 25-28). At the time applicant's invention was made, it would have been obvious to one skilled in the art to further modify Van Dyke's invention such that the attempter was a kernel mode process. The rationale for why it would have been obvious is that substituting a kernel mode process for Van Dyke's process is simple substitution of one known element (used for similar purposes) for another to obtain predictable results.

### ***Conclusion***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to PONNOREAY PICH whose telephone number is (571)272-7962. The examiner can normally be reached on 9:00am-4:30pm Mon-Thurs.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kim Vu can be reached on 571-272-3859. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Ponnoreay Pich/
Examiner, Art Unit 2135